$AR

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the Application of: Charisius et al.
Application No: 09/839,524
Filed: 04/20/2001
Group Art Unit: 2162
Examiner: Nahar, Qamrun

For: **METHODS AND SYSTEMS FOR RELATING A DATA DEFINITION FILE AND A DATA MODEL FOR DISTRIBUTED COMPUTING**

APPEAL BRIEF

Board of Patent Appeals and Interferences
U.S. Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

Sirs:

This Appeal Brief is pursuant to the Notice of Appeal filed on April 3, 2006. The application is on behalf of a large entity. A check for $500 covering the fee for the Appeal Brief is enclosed. The Commissioner is authorized to charge any additional fees that may be required to Deposit Account 501923. This brief contains these items under the following headings, and in the order set forth below:

1. Real Party in Interest

The real party in interest in this appeal is Borland Software Corporation, the parent corporation of Togethersoft, the assignee of record.

2. Related Appeals and Interferences

There are no related Appeals and Interferences.

3. Status of the Claims

Claims 1-70 are pending. Claims 1, 9, 19, 24, 32, 40, 50, 55, 63, and 70 are independent claims.

4. Status of Amendments

No claims were amended subsequent the Examiner's Final Rejection.

5. Summary of Claimed Subject Matter

Methods and systems consistent with the present invention provide an improved software development tool that overcomes the limitations of conventional software development tools. The improved software development tool of the present invention allows a developer to simultaneously view a graphical and a textual display of source code. The graphical and textual views are synchronized so that a modification in one view is automatically reflected in the other view. The software development tool is designed for use with more than one programming language.

The software development tool significantly reduces programming development time for a developer by allowing the developer to generate a data model, such as an Extensible Markup Language (XML) structure diagram, from a definition file, such as a Document Type Definition (DTD) file. The XML structure diagram produced by the software development tool provides the developer with a visual insight into the design of the data definition file so that problems with the data definition file or code using the data definition file can be corrected quickly. In addition, the software development tool saves the developer the time and effort spent manually producing a DTD file by allowing the developer to automatically generate a DTD from an XML structure diagram previously produced by the developer using the software development tool.

In accordance with methods consistent with the present invention, a method is provided in a data processing system. The method includes the steps of receiving an identification of a data definition file having a plurality of data elements and a plurality of relationships between the data elements, each data element having a name and a definition, and generating a graphical

representation that visually identifies the plurality of data elements and the plurality of relationships between the data elements contained within the data definition file.

In accordance with methods consistent with the present invention, a method is provided in a data processing system. The method includes the steps of receiving an indication to generate a data definition file from a graphical representation having a plurality of data element diagrams and a plurality of relationships between the data element diagrams, each data element diagram having a name, and adding a plurality of data element identifiers to the data definition file that reflect the data element diagrams and the relationships between the data element diagrams.

In accordance with articles of manufacture consistent with the present invention, a computer-readable medium is provided. The computer-readable medium contains instructions for controlling a data processing system to perform one of the other of the methods just described.

6. Grounds of Rejection to be Reviewed on Appeal

The two grounds of rejection are:

1. Claims 1, 5-9, 12-19, 23-24, 27-32, 36-40, 43-50, 54-55, 58-63, 66-67 and 70 stand rejected under 35 U.S.C. 102(e) as being anticipated by Carpenter-Smith (U.S. 5,838,973). Claims 1, 5-9, 12-19, 23-24, 27-32, 36-40, 43-50, 54-55, 58-63, 66-67 and 70 can be grouped together.

2. Claims 2-4, 10-11, 20-22, 25-26, 33-35, 41-42, 51-53, 56-57, 64-65 and 68-69 stand rejected under 35 U.S.C. 103(a) as being unpatenable over Carpenter-Smith (U.S. 5,838,973) in view of Mutschler (U.S. 6,381,743). Claims 2-4, 10-

11, 20-22, 25-26, 33-35, 41-42, 51-53, 56-57, 64-65 and 68-69 can be grouped together.

7. Argument

7.1: Claims 1, 5-9, 12-19, 23-24, 27-32, 36-40, 43-50, 54-55, 58-63, 66-67 and 70 stand rejected under 35 U.S.C. 102(e) as being anticipated by Carpenter-Smith (U.S. 5,838,973).

The Present Invention Is Novel Over Carpenter-Smith (U.S. 5,838,973)

As stated in MPEP §2131, a claim is anticipated under §102 only if each and every element as set forth in the claim, in as complete of detail, is found in a single prior art reference. The claimed invention, according to independent claim 1, includes a recitation for a step of synchronizing a textual relationship with a graphical relationship so that a modification in one representation is automatically visible in the other representation. As such, for Carpenter-Smith to be anticipatory, it must describe this identical element. Carpenter-Smith does not teach the step of synchronizing a textual relationship with a graphical relationship so that a modification in one representation is automatically visible in the other representation, or its advantages, as explained below.

In the 09 January 2006 Final Office Action, the Examiner asserted that U.S. Patent No. 5,838,973 to Carpenter-Smith teaches synchronizing a textual relationship with a graphical relationship so that a modification in one representation is automatically visible in the other representation. The Examiner heavily relies on Fig. 19 of the Carpenter-Smith reference, pointing out that "FIG. 19 shows the recipient list window 350, having a responsibility list 352, a class list 354 and a behavior name list 356. When the user selects the yes button 344 in the action... As seen in FIG. 19, **the 2D object window 270 can be shown simultaneously with the**

recipient list window 350 to assist the user in defining the classes and behaviors." citing column 12, lines 32-45, emphasis added.

Applicant's counsel does not see how the Examiner's emphasis relates to a teaching of synchronizing a textual relationship with a graphical relationship so that a modification in one representation is automatically visible in the other representation as currently called for in the independent claims of the present application.

FIG. 19 of Carpenter-Smith merely illustrates a windowing feature whereby a user can simultaneously display the recipient list window 350 for review in order to make it easier for a user to interactively construct graphical representations of objects in a simultaneously displayed two-dimensional object window 270. In other words, as depicted in the drawings of Carpenter-Smith and described in the specification of Carpenter-Smith, a user can refer to the textual representations of window 350 while using a graphical user interface to construct and/or modify the graphical representations of the 2D object window 270.

In order to anticipate the present invention as currently claimed, Carpenter-Smith would have to show an automatic synchronization between recipient window 350 and the 2D object window 270. Carpenter-Smith does not disclose that a modification in either the recipient list window 350 or the 2D object window 270 is automatically visible in the other window. Therefore, Carpenter-Smith does not disclose synchronizing a textual relationship with a graphical relationship so that a modification in one representation is automatically visible in the other representation as currently called for in the independent claims of the present application.

Carpenter-Smith does automatically enter user tagged action phrases into the responsibility list 352, but this action does not perform a step of synchronization between graphical and textual representations, because tagged phrases are not graphical representations.

Thus, unlike the present invention's method automatic synchronization between graphical and textual representations, Carpenter-Smith relies upon the user to maintain synchronization between a textual representation in recipient window 350 and a graphical representation in the 2D object window 270.

Carpenter-Smith gives the user a graphical user interface tool with which the user interacts to maintain synchronization between recipient window 350 and the 2D object window 270. This user maintenance of synchronization disclosed by Carpenter-Smith is not automatic. Please notice the 2D object tool bar shown in the upper right-hand corner of window 270 shown in FIG. 19. This 2D object tool bar is labeled 272 in FIG. 13 of Carpenter-Smith and includes a pointer object tool 274 that allows the user to move 2D objects from one location in 2D object window 270 to another location in 2D object window 270. Object tool bar 272 also includes a new object tool 278 that, when selected, allows a user to add new objects to 2D object window 270 and a text object tool 276 for labeling objects created by the user.

Moreover, object tool bar 272 of Carpenter-Smith includes a link object tool that allows the user to draw lines connecting certain objects to show a relation between those connected objects. The graphical user interface 2D object tool 272 of Carpenter-Smith is not needed by the present invention to maintain synchronization between a textual representation and a graphical representation, because the present invention includes a step that automatically synchronizes a textual relationship with a graphical relationship so that a modification in one representation is automatically visible in the other representation. If Carpenter-Smith included the claimed steps of the present invention, there would be no need for the 2D object tool shown in FIG. 19 of Carpenter-Smith on which the Examiner relies in making his rejection. The windowing feature disclosed by Carpenter-Smith as depicted in FIG. 19 and described in column 12, lines 32-45 in

no way suggests automatic synchronization between a textual representation and a graphical representation as called for in the Applicant's independent claims. Therefore, the Carpenter-Smith patent cannot anticipate the present invention as claimed. As a result, Applicants respectfully request that the rejection of claims 1, 5-9, 12-19, 23-24, 27-32, 36-40, 43-50, 54-55, 58-63, 66-67 and 70 be reversed.

7.2: Claims 2-4, 10-11, 20-22, 25-26, 33-35, 41-42, 51-53, 56-57, 64-65 and 68-69 stand rejected under 35 U.S.C. 103(a) as being unpatenable over Carpenter-Smith (U.S. 5,838,973) in view of Mutschler (U.S. 6,381,743).

The Present Invention Would not have Been Obvious from The Cited References

A claimed invention may be found to have been obvious "if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains." 35 U.S.C. § 103(a). Moreover, the Federal Circuit has ruled on numerous occasions that a holding of "obviousness" requires some motivation, suggestion or teaching within the cited references that would lead one skilled in the art to modify the cited reference or references as claimed by applicant. See, for example, *In re Kotzab*, 217 F3d 1365, 55 USPQ2d 1313 (Fed Cir. 2000):

> "Most if not all inventions arise from a combination of old elements. See *In re Rouffet*, 149 F.3d 1350, 1357, 47 USPQ2d 1453, 1457 (Fed. Cir. 1998). Thus, every element of a claimed invention may often be found in the prior art. However, identification in the prior art of each individual part claimed is insufficient to defeat patentability of the whole claimed invention. Rather, to establish obviousness based on a combination of the elements disclosed in the prior art, there must be some motivation, suggestion or teaching of the desirability of making the specific combination that was made by the applicant. See *In re Dance*, 160 F.3d 1339, 1343, 48 USPQ2d 1635, 1637 (Fed. Cir. 1998); *In re Gordon*, 733 F.2d 900, 902, 221 USPQ 1125, 1127 (Fed. Cir. 1984). Even when obviousness is based on a single prior art reference, there must be a showing of a

suggestion or motivation to modify the teachings of that reference. See *B.F. Goodrich Co. v. Aircraft Breaking Sys. Corp.*, 72 F.3d 1577, 1582, 37 USPQ2d 1314, 1318 (Fed. Cir. 1996)."

The invention described by Carpenter-Smith is a software tool that guides a user through an object-oriented software development process. The invention described by Mutschler is a software tool useful for interchanging metadata between modeling tools and metadata repositories in distributed heterogeneous environments. The present invention also discloses a software tool. However, the software tool of the present invention as specifically claimed is novel and non-obvious over the cited references.

Specifically, Carpenter-Smith fails to disclose the claimed method of the present invention because the present method includes a step of synchronizing a textual representation with the graphical representation so that a modification in one representation is automatically visible in the other representation.

Conventional software development tools such as Carpenter-Smith allow a programmer to view a graphical representation such as UML while viewing a textual representation of source code. However, with most ordinary software development tools, the source code is stored in a file, and a reverse engineering module converts the source code into a representation of the software project in a database or repository. A software project being developed using a software development tool comprises source code in at least one file which, when compiled, forms a sequence of instructions to be run by the data processing system. The repository generates the graphical representation, which is normally UML. If any changes are made to the graphical representation, they are automatically reflected in the repository, and a code generator converts the representation in the repository into source code. Such conventional software development tools, however, do not synchronize the displays of the graphical representation and

textual representations of the source code. Rather, the repository typically stores the graphical representation of the software project while the file stores the textual representation. A modification in the graphical representation does not appear in the textual representation unless the code generator re-generates the source code from the data in the repository. When this occurs, the entire source code (i.e., the textual representation) is rewritten. Similarly, any modifications made to the source code do not appear in the graphical representation unless the reverse engineering module updates the repository. As a result, redundant information is stored in the repository and the source code. In addition, rather than making incremental changes to the source code, conventional software development tools rewrite the overall source code when modifications are made to the graphical representation, resulting in wasted processing time. This type of manual, large-grained synchronization requires either human intervention such as disclosed by Carpenter-Smith, or by a "batch" style process to try to keep the two views (the graphical representation and the source code) in synchronization. Unfortunately, this approach, adopted by many tools including the cited references, leads to many undesirable side-effects, such as desired changes to the source code being overwritten by the tool.

The improved software development tool of the present invention allows a developer to simultaneously view graphical and textual views that are automatically synchronized by the present computerized method so that a modification in one view is automatically reflected in the other view. Neither Carpenter-Smith nor Mutschler automatically synchronize graphical and textual representations of source code. Moreover, there is no suggestion, teaching or motivation for combining Mutschler with Carpenter-Smith to come up with the present invention as claimed. The references cited but not applied, alone or in combination, fail to contain any teaching or suggestion of a method that includes a step of synchronizing a textual representation

with the graphical representation so that a modification in one representation is automatically visible in the other representation. As a result, Applicants respectfully request that the rejection of claims 2-4, 10-11, 20-22, 25-26, 33-35, 41-42, 51-53, 56-57, 64-65 and 68-69 be reversed.

7.3: Conclusion

The Examiner's rejections of Claims 1-70 should be reversed.

8. Claims Appendix

Listing of Appealed Claims

1. A computerized method in a data processing system, the method comprising the steps of:

receiving a request to automatically generate a data model from a data definition file containing a data structure with a plurality of data elements, each data element having a name and a definition;

generating a diagram for each data element, the diagrams having names corresponding to the data elements;

determining whether an attribute in the data structure is associated with one of the data elements;

when it is determined that an attribute in the data structure is associated with one of the data elements,

displaying a graphical representation of the attribute with the diagram corresponding to the one data element;

synchronizing a textual representation with the graphical representation so that a modification in one representation is automatically visible in the other representation;

determining whether the data structure includes a reference between two of the data elements;

when it is determined that the data structure includes a reference between two elements,

displaying a reference link from the diagram associated with a first of the two elements to the diagram associated with a second of the two elements;

determining whether the data structure includes a group attribute identifying at least one of the data elements as group attribute member;

when it is determined that the data structure includes a group attribute,

displaying a group symbol;

displaying a first link between the group symbol and the diagram that is associated with the identified at least one data element; and

displaying a second link between the group symbol and the diagram of the data element having the associated group attribute.

2. The method of claim 1, wherein the data model is an XML structure diagram.

3. The method of claim 1, wherein the data definition file is a Document Type Definition file.

4. The method of claim 1, wherein the data definition file is an XML schema file.

5. The method of claim 1, wherein the step of determining whether the data structure includes a reference between two of the data elements comprises the step of determining whether the name of the second element is within the definition of the first data element.

6. The method of claim 1, wherein the step of displaying the second link further comprises the step of determining whether the data element has the associated group attribute.

7. The method of claim 6, wherein the step of determining whether the data element has the associated group attribute comprises the step of determining whether the group attribute is within the definition of the data element.

8. The method of claim 1, further comprising the step of storing information used to generate the data model in a graphical view file.

9. A computerized method in a data processing system comprising the steps of:

receiving an identification of a data definition file having a plurality of data elements and a plurality of relationships between the data elements, each data element having a name and a definition; and

generating a graphical representation that visually identifies the plurality of data elements and the plurality of relationships between the data elements contained within the data definition file, wherein a textual representation is synchronized with the graphical representation so that a modification in one representation is automatically visible in the other representation.

10. The method of claim 9, wherein the graphical representation corresponds to an XML structure diagram.

11. The method of claim 9, wherein the data definition file corresponds to a Document Type Definition file.

12. The method of claim 9, wherein the step of generating a graphical representation comprises the step of generating a diagram for each data element, the diagrams having names corresponding to the data elements.

13. The method of claim 12, wherein the step of generating a graphical representation further comprises the steps of:

determining whether the data structure contains an attribute for one of the data elements; and

when it is determined that the data structure contains an attribute for one of the data elements,

displaying a graphical representation of the attribute with the diagram corresponding to the one data element.

14. The method of claim 12, wherein the step of generating a graphical representation further comprises the steps of:

identifying a first of the relationships as an element reference between two of the data elements; and

displaying a reference link from the diagram associated with a first of the two elements to the diagram associated with a second of the two elements.

15. The method of claim 14, wherein the step of identifying a first of the relationships comprises the step of determining whether the name of the second element is within the definition of the first data element.

16. The method of claim 12, wherein the step of generating a graphical representation further comprises the steps of:

identifying a second of the relationships as a group reference between two of the data elements;

displaying a group symbol; displaying a first link between the group symbol and the diagram that is associated with a first of the two elements; and

displaying a second link between the group symbol and the diagram that is associate with a second of the two elements.

17. The method of claim 16, wherein the step of identifying a second of the relationships comprises the steps of:

determining whether the data structure includes a group attribute identifying the first data element as a group attribute member; and

determining whether the definition of the second data element contains the group attribute.

18. The method of claim 9, further comprising the step of storing information used to generate the data model in a graphical view file.

19. A computerized method in a data processing system, the method comprising the steps of:

receiving a request to automatically generate a data definition file from a graphical representation having a plurality of data element diagrams, each data element diagram having a name;

synchronizing a textual representation with the graphical representation so that a modification in one representation is automatically visible in the other representation;

adding a data element identifier in the data definition file for each data element diagram in the graphical representation;

adding the name of each data element diagram to the corresponding data element identifier;

determining whether an attribute is associated with one of the data element diagrams;

when it is determined that an attribute is associated with one of the data element diagrams,

adding the attribute to the data element identifier for the one data element diagram;

determining whether the graphical representation has a reference link between two of the data element diagrams;

when it is determined that the graphical representation has a reference link between two data element diagrams,

adding a reference to the data element identifier corresponding to one of the two data element diagrams to reflect a link between the two data element diagrams;

determining whether the graphical representation has a group symbol;

when it is determined that the graphical representation has a group symbol,

determining whether the group symbol is a destination in a first link with a first of the data element diagrams;

when it is determined that the group symbol is a destination in a first link with a first of the data element diagrams,

adding a group definition to the data definition file that includes the name of the first data element diagram;

determining whether the group symbol is a source in a second link with a second of the data element diagrams; and

when it is determined that the group symbol is a source in a second link with a second of the data element diagrams,

adding to the data element identifier corresponding to the second element diagram a reference corresponding to the group definition.

20. The method of claim 19, wherein the graphical representation corresponds to an XML structure diagram.

21. The method of claim 19, wherein the data definition file corresponds to a Document Type Definition file.

22. The method of claim 19, wherein the data definition file corresponds to an XML schema file.

23. The method of claim 19, wherein the step of adding a reference to the data element identifier comprises the steps of:

identifying one of two data element diagrams as a source of the reference link and the other as a destination of the reference link; and

adding the name of the destination to the data element identifier corresponding to the source.

24. A computerized method in a data processing system comprising the steps of:

receiving an indication to automatically generate a data definition file from a graphical representation having a plurality of data element diagrams and a plurality of relationships between the data element diagrams, each data element diagram having a name;

synchronizing a textual representation with the graphical representation so that a modification in one representation is automatically visible in the other representation; and

adding a plurality of data element identifiers to the data definition file that reflect the data element diagrams and the relationships between the data element diagrams.

25. The method of claim 24, wherein the graphical representation corresponds to an XML structure diagram.

26. The method of claim 24, wherein the data definition file corresponds to a Document Type Definition file.

27. The method of claim 24, wherein the step of adding the plurality of data element identifiers to the data definition file comprises the step of adding the name of each data element diagram to the corresponding data element identifier.

28. The method of claim 24, wherein the step of adding the plurality of data element identifiers to the data definition file comprises the steps of:

determining whether an attribute is associated with one of the data element diagrams;

when it is determined that an attribute is associated with one of the data element diagrams, and

adding the attribute to the data element identifier for the one data element diagram.

29. The method of claim 24, wherein the step of adding the plurality of data element identifiers to the data definition file comprises the steps of:

determining whether one of the relationships corresponds to a reference link between two of the data element diagrams; and

when it is determined that one of the relationships corresponds to a reference link between two data element diagrams,

adding a reference to the data element identifier corresponding to one of the two data element diagrams to reflect the reference link between the two data element diagrams.

30. The method of claim 24, wherein the step of adding the plurality of data element identifiers to the data definition file comprises the steps of:

determining whether another of the relationships is associated with a group symbol;

when it is determined that the other of the relationships is associated with a group symbol,

determining whether the group symbol is a destination in a first link with a first of the data element diagrams; and

when it is determined that the group symbol is a destination in a first link with a first of the data element diagrams,

adding a group definition to the data definition file that includes the name of the first data element diagram.

31. The method of claim 30, further comprising the steps of:

when it is determined that the group symbol is a destination in a first link with a first of the data element diagrams,

determining whether the group symbol is a source in a second link with a second of the data element diagrams; and

when it is determined that the group symbol is a source in a second link with a second of the data element diagrams,

adding to the data element identifier corresponding to the second element diagram a reference corresponding to the group definition.

32. A computer-readable medium containing software instructions for controlling a data processing system to perform a method, the method comprising the steps of:

receiving a request to automatically generate a data model from a data definition file containing a data structure with a plurality of data elements, each data element having a name and a definition;

generating a diagram for each data element, the diagrams having names corresponding to the data elements;

determining whether an attribute in the data structure is associated with one of the data elements;

when it is determined that an attribute in the data structure is associated with one of the data elements,

displaying a graphical representation of the attribute with the diagram corresponding to the one data element;

synchronizing a textual representation with the graphical representation so that a modification in one representation is automatically visible in the other representation;

determining whether the data structure includes a reference between two of the data elements;

when it is determined that the data structure includes a reference between two elements,

displaying a reference link from the diagram associated with a first of the two elements to the diagram associated with a second of the two elements;

determining whether the data structure includes a group attribute identifying at least one of the data elements as group attribute member;

when it is determined that the data structure includes a group attribute,

displaying a group symbol;

displaying a first link between the group symbol and the diagram that is associated with the identified at least one data element; and

displaying a second link between the group symbol and the diagram of the data element having the associated group attribute.

33. The computer-readable medium of claim 32, wherein the data model is an XML structure diagram.

34. The computer-readable medium of claim 32, wherein the data definition file is a Document Type Definition file.

35. The computer-readable medium of claim 32, wherein the data definition file is an XML schema file.

36. The computer-readable medium of claim 32, wherein the step of determining whether the data structure includes a reference between two of the data elements comprises the step of determining whether the name of the second element is within the definition of the first data element.

37. The computer-readable medium of claim 32, wherein the step of displaying the second link further comprises the step of determining whether the data element has the associated group attribute.

38. The computer-readable medium of claim 37, wherein the step of determining whether the data element has the associated group attribute comprises the step of determining whether the group attribute is within the definition of the data element.

39. The computer-readable medium of claim 32, wherein the method further comprises the step of storing information used to generate the data model in a graphical view file.

40. A computer-readable medium containing software instructions for controlling a data processing system to perform a method, the method comprising the steps of:

receiving an identification of a data definition file having a plurality of data elements and a plurality of relationships between the data elements, each data element having a name and a definition; and

generating a graphical representation that visually identifies the plurality of data elements and the plurality of relationships between the data elements contained within the data definition file, wherein a textual representation is synchronized with the graphical representation so that a modification in one representation is automatically visible in the other representation.

41. The computer-readable medium of claim 40, wherein the graphical representation corresponds to an XML structure diagram.

42. The computer-readable medium of claim 40, wherein the data definition file corresponds to a Document Type Definition file.

43. The computer-readable medium of claim 40, wherein the step of generating a graphical representation comprises the step of generating a diagram for each data element, the diagrams having names corresponding to the data elements.

44. The computer-readable medium of claim 43, wherein the step of generating a graphical representation further comprises the steps of:

determining whether the data structure contains an attribute for one of the data elements; and

when it is determined that the data structure contains an attribute for one of the data elements,

displaying a graphical representation of the attribute with the diagram corresponding to the one data element.

45. The computer-readable medium of claim 43, wherein the step of generating a graphical representation further comprises the steps of:

identifying a first of the relationships as an element reference between two of the data elements; and

displaying a reference link from the diagram associated with a first of the two elements to the diagram associated with a second of the two elements.

46. The computer-readable medium of claim 45, wherein the step of identifying a first of the relationships comprises the step of determining whether the name of the second element is within the definition of the first data element.

47. The computer-readable medium of claim 43, wherein the step of generating a graphical representation further comprises the steps of:

identifying a second of the relationships as a group reference between two of the data elements;

displaying a group symbol;

displaying a first link between the group symbol and the diagram that is associated with a first of the two elements; and

displaying a second link between the group symbol and the diagram that is associate with a second of the two elements.

48. The computer-readable medium of claim 47, wherein the step of identifying a second of the relationships comprises the steps of:

determining whether the data structure includes a group attribute identifying the first data element as a group attribute member; and

determining whether the definition of the second data element contains the group attribute.

49. The computer-readable medium of claim 40, wherein the method further comprises the step of storing information used to generate the data model in a graphical view file.

50. A computer-readable medium containing software instructions for controlling a data processing system to perform a method, the method comprising the steps of:

receiving a request to automatically generate a data definition file from a graphical representation having a plurality of data element diagrams, each data element diagram having a name;

synchronizing a textual representation with the graphical representation so that a modification in one representation is automatically visible in the other representation;

adding a data element identifier in the data definition file for each data element diagram in the graphical representation;

adding the name of each data element diagram to the corresponding data element identifier;

determining whether an attribute is associated with one of the data element diagrams;

when it is determined that an attribute is associated with one of the data element diagrams,

adding the attribute to the data element identifier for the one data element diagram;

determining whether the graphical representation has a reference link between two of the data element diagrams;

when it is determined that the graphical representation has a reference link between two data element diagrams,

adding a reference to the data element identifier corresponding to one of the two data element diagrams to reflect a link between the two data element diagrams;

determining whether the graphical representation has a group symbol;

when it is determined that the graphical representation has a group symbol,

determining whether the group symbol is a destination in a first link with a first of the data element diagrams;

when it is determined that the group symbol is a destination in a first link with a first of the data element diagrams,

adding a group definition to the data definition file that includes the name of the first data element diagram;

determining whether the group symbol is a source in a second link with a second of the data element diagrams; and

when it is determined that the group symbol is a source in a second link with a second of the data element diagrams,

adding to the data element identifier corresponding to the second element diagram a reference corresponding to the group definition.

51. The computer-readable medium of claim 50, wherein the graphical representation corresponds to an XML structure diagram.

52. The computer-readable medium of claim 50, wherein the data definition file corresponds to a Document Type Definition file.

53. The computer-readable medium of claim 50, wherein the data definition file corresponds to an XML schema file.

54. The computer-readable medium of claim 50, wherein the step of adding a reference to the data element identifier comprises the steps of:

identifying one of two data element diagrams as a source of the reference link and the other as a destination of the reference link; and

adding the name of the destination to the data element identifier corresponding to the source.

55. A computer-readable medium containing software instructions for controlling a data processing system to perform a method, the method comprising the steps of:

receiving an indication to automatically generate a data definition file from a graphical representation having a plurality of data element diagrams and a plurality of relationships between the data element diagrams, each data element diagram having a name;

synchronizing a textual representation with the graphical representation so that a modification in one representation is automatically visible in the other representation; and

adding a plurality of data element identifiers to the data definition file that reflect the data element diagrams and the relationships between the data element diagrams.

56. The computer-readable medium of claim 55, wherein the graphical representation corresponds to an XML structure diagram.

57. The computer-readable medium of claim 55, wherein the data definition file corresponds to a Document Type Definition file.

58. The computer-readable medium of claim 55, wherein the step of adding the plurality of data element identifiers to the data definition file comprises the step of adding the name of each data element diagram to the corresponding data element identifier.

59. The computer-readable medium of claim 55, wherein the step of adding the plurality of data element identifiers to the data definition file comprises the steps of:

determining whether an attribute is associated with one of the data element diagrams; and

when it is determined that an attribute is associated with one of the data element diagrams,

adding the attribute to the data element identifier for the one data element diagram.

60. The computer-readable medium of claim 55, wherein the step of adding the plurality of data element identifiers to the data definition file comprises the steps of:

determining whether one of the relationships corresponds to a reference link between two of the data element diagrams; and

when it is determined that one of the relationships corresponds to a reference link between two data element diagrams,

adding a reference to the data element identifier corresponding to one of the two data element diagrams to reflect the reference link between the two data element diagrams.

61. The computer-readable medium of claim 55, wherein the step of adding the plurality of data element identifiers to the data definition file comprises the steps of:

determining whether another of the relationships is associated with a group symbol;

when it is determined that the other of the relationships is associated with a group symbol,

determining whether the group symbol is a destination in a first link with a first of the data element diagrams; and

when it is determined that the group symbol is a destination in a first link with a first of the data element diagrams,

adding a group definition to the data definition file that includes the name of the first data element diagram.

62. The computer-readable medium of claim 61, wherein the method further comprises the steps of:

when it is determined that the group symbol is a destination in a first link with a first of the data element diagrams,

determining whether the group symbol is a source in a second link with a second of the data element diagrams; and

when it is determined that the group symbol is a source in a second link with a second of the data element diagrams,

adding to the data element identifier corresponding to the second element diagram a reference corresponding to the group definition.

63. A data processing system comprising:

a secondary storage device comprising a first data definition file having first data elements and relationships between the first data elements;

a memory device further comprising a program that receives a first request to

display a first graphical representation of the first data definition file such that the first graphical representation has first data element diagrams and relationships between the first data element diagrams reflecting the first data elements and the relationships between the first data elements contained in the first data definition file, that displays the first graphical representation responsive to receiving the first request, that displays a second graphical representation having second data element diagrams and relationships between the second data element diagrams, that receives a second request to automatically generate a second data definition file from the second graphical representation such that the second data definition file has second data elements and relationships between the second data elements reflecting the second data element diagrams and the relationships between the second data element diagrams of the second graphical representation, and that automatically generates the second data definition file responsive to receiving the second request; that synchronizes a textual representation with the graphical

representations so that a modification in each representation is automatically visible in the other representation; and

a processor for running the program.

64. The data processing system of claim 63, wherein the first data definition file corresponds to a Document Type Definition file.

65. The data processing system of claim 63, wherein the second data definition file corresponds to a Document Type Definition file.

66. The data processing system of claim 63, wherein the program further stores the second graphical representation in a second graphical view file on the secondary storage device.

67. The data processing system of claim 63, wherein the program further stores the second data definition file on the secondary storage device.

68. The data processing system of claim 63, wherein the secondary storage device further comprises an XML data structure module that includes XML constructs corresponding to a Document Type Definition specification.

69. The data processing system of claim 68, wherein the program uses the XML constructs to add the plurality of data element identifiers to the first data definition file and to parse the plurality of data elements contained in the second data definition file.

70. A system comprising:

means for receiving an identification of a data definition file having a plurality of data elements and a plurality of relationships between the data elements, each data element having a name and a definition;

means for automatically generating a graphical representation that visually identifies the plurality of data elements and the plurality of relationships between the data elements contained within the data definition file; and

means for synchronizing a textual representation with the graphical representation so that a modification in one representation is automatically visible in the other representation.

9. Evidence Appendix

These references were cited by the Examiner in his rejections, and applicant relies on portions of them to show the errors of the rejections. Copies are attached.

Patent Number or Document Number	1st Named Inventor	Examiner Cited in Office Action Dated
U.S. 5,838,973	Carpenter-Smith	January 9, 2006
U.S. 6,381,743	Mutschler	January 9, 2006

10. Related Proceedings Appendix

None.

Respectfully submitted,

Howard A. MacCord, Jr.
Registration No. 28,639
MacCord Mason PLLC
P. O. Box 2974
Greensboro, North Carolina 27402
(336) 273-4422.

Date: May 25, 2006
File No.: 7399-023

CERTIFICATE OF MAILING

I HEREBY CERTIFY THAT THIS DOCUMENT IS BEING DEPOSITED WITH THE UNITED STATES POSTAL SERVICE AS FIRST-CLASS MAIL, IN AN ENVELOPE ADDRESSED TO: **COMMISSIONER FOR PATENTS, P.O. BOX 1450, ALEXANDRIA, VA 22313-1450**, ON 25 MAY 2006
(Date of Deposit)

Christian E. Carter-Seyboth
Name of Depositor

Christian E. Carter-Seyboth
Signature

25 MAY 2006
Date of Signature